

**SPECIAL
OFFER!**

THE DISCIPLE DISC-MANAGER

**A SUITE OF INTERACTIVE PROGRAMS
TO MAKE THE MOST OF YOUR DISCIPLE**

GET TOTAL CONTROL OF YOUR DISCS

COMPRISES **3** PROGRAMS

SPECIAL INDUG PRICE

ONLY **£9.95**
SAVE £5

The DISCIPLE Disc Manager gives you powerful disc handling and organising capabilities that would cost £100's on other machines!

● **THE CONFIGURER** This program actually names and numbers DISCIPLE discs to your requirements. Now disc identification is simplicity itself!

● **THE CATALOGUER** An amazing program that allows you to catalogue, search, update and display every program on every disc... **NO** typing in, simply insert all your discs, one at a time into drive 1 and the Cataloguer sorts, collates and displays all programs into numerical order, **automatically!**

● **THE AUTOLOADER** This program is the real workhorse of this suite of programs. You can **automatically** load, erase, copy, re-name all programs with 2 catalogue display options, and 2 disc facility. You can now **ERASE** or **RE-NAME** up to **15** programs at **one time**. (Ideal for multiple re-naming after a snap shot session). The Autoloader on every disc makes using the DISCIPLE so simple, so easy and such a pleasure to use, that you'll wonder how you could ever have Managed without it!

**COMPATIBLE
WITH THE NEW
★ PLUS D ★**

ONLY
AVAILABLE
FROM

BETTER BYTES

BETTERBYTES, 10 SPITAL TERRACE, GOSFORTH, NEWCASTLE UPON TYNE NE3 1UT

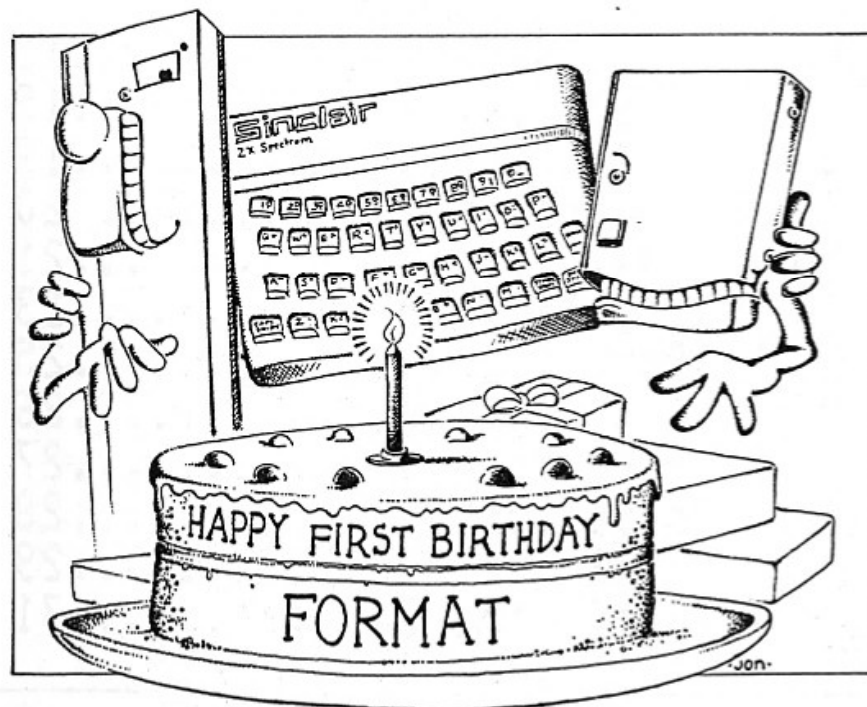
Mail Orders Welcome State if Disciple or Plus D and give details of drive type etc. Overseas orders add £1.50 p&p
(Please make cheques payable to D.L. HOOD)

Vol 2 - No 1.

August 1988.

FORMAT

THE MONTHLY MAGAZINE FOR
DISCIPLE & PLUS D USERS



SPECIAL BIRTHDAY ISSUE

INDUG.

CONTENTS

The Editor Speaks.....	3
News On 4.....	4
Your Letters.....	5
Greetings From Swansea.....	6
Byting A Bit Less.....	9
The Help Page.....	12
Me & My Computer.....	13
SIDEWRITER.....	15
DUMPY - A Review.....	19
A (fun) Glossary.....	20
Hack Zone.....	21
Expanding Tasword 2.....	23
Table Fun.....	25
Small Ads.....	26
Tasword +2 Pokes.....	27
Back Issue Service.....	28
Teletext Explained.....	29
Format Software Service.....	31

(C)Copyright 1988 INDUG. All Rights Reserved.

No part of this publication may be reproduced, in any form, without the written consent of the publisher. INDUG members may copy program material only for their own personal use.

FORMAT is published by INDUG, 34 Bourton Road, Gloucester, GL4 0LE, England. Telephone 0452-412572. DISCiPLE and PLUS D are trade marks of MILES GORDON TECHNOLOGY, Lakeside Technology Park, Phoenix Way, Swansea, South Wales, SA7 9EH. The DISCiPLE interface is marketed by Rockfort Products, 81 Church Road, Hendon, London, NW4 4DP.



Our FIRST BIRTHDAY issue, I never thought we would get this far. If I had known how big things would get I would have laid better plans before getting started. We now have OVER 1000 members, thats about 4 or 5 times what I planned for and I have to spend a good deal of my time stretching the computer systems to cope with everyone. But I enjoy it, and thats the main thing.

My thanks to everyone who wrote birthday messages, I read all your letters even if there isn't time to reply to everyone. Its nice to hear your comments on FORMAT, whether good or bad, keep them coming.

Twelve months ago, as I put together issue #1 of FORMAT we already had members as far away as Australia and New Zealand, well for those of you who are interested here is the current list:- Australia; Austria; Belgium; Canada; Cote D'Ivoire; Denmark; Finland; France; Greece; Guinea Bissau; Hungary; India; Ireland; Israel; Italy; Luxembourg; Malawi; Malta; New Zealand; Nigeria; Norway; Portugal; Saudi Arabia; South Africa; Spain; Sultanate of Oman; Sweden; Switzerland; The Netherlands; Turkey; U.S.A.; West Germany; West Malaysia; Yugoslavia; and Zambia, with apologies to any I have missed. As you see we cover the world very well, around 16% of our members are overseas, but we also cover the length and breadth of Britain.

My thanks go to the many writers and programmers who have contributed to FORMAT over the last twelve months. Keep up the good work. I would also like to thank Alan Miles and Bruce Gordon for their help, encouragement and support. A special thanks must go to Bruce for designing such wonderful interfaces, I sometime think he has single handedly saved the Spectrum from a premature death.

I look forward to the next twelve months with great pleasure, FORMAT is growing in size and scope. Our readership is growing all the time. This month we welcome Clyde Bish to our pages, Clyde wrote for ZX Computing (until it disappeared last year) will be a regular contributor in the future. Over the next few months we have new series starting on Adventures; Midi Interfacing; Machine Code Programming and Advanced Disc Programming.

Finally, the success of FORMAT is down to you, the readers, without whom there would be no point in writing, THANK YOU.

See you next month.

Bob Brenchley. Editor.

NEWS ON 4

RAM PRICES SET TO FALL

Did you know there was a futures market for RAM chips? Well there is in the United States. It seems chips are now so important that they take their place along side Grain, Beef and Pork Bellies in the New York markets and after 14 months of constant rises the price is beginning to fall. OK only by a few cents, but this could herald the end of the shortages and high prices which have bedevilled the industry over the last twelve months.

HACKING AROUND

Games hackers seem to be onto a winning streak this month with news of new packages aimed at them. MGT have announced Pick-POKE-It for the PLUS D, a software package to give on-line features like Disassembly, Memory examination and searching, POKEing etc. INDUG has also launched PLUS D HACKER by Steve Nutting which includes features for Graphic/Screen searching and a set of extensions to basic for Graphics Animation (see P.S.S. page 31). We'll bring you a comparative review as soon as we can.

DISC SALES

A recent survey published in trade magazines predicts that sales of 3.5" disc will match those of 5.25" ones by 1991. With more Business machines, like the new IBM PS/2 micros, adopting the 3.5" as standard sales are fast catching up. Even so sales of 5.25" discs will still continue to rise as the total market expands. The survey predicts sales of around 1300 million 5.25" discs this year with 3.5" selling about 425 million. 8" and 3" form such a small part of the market that they dropped off the bottom of the chart.

EEC PRINTER LEVY LATEST

The European Economic Community continue to wage war on what they believe is the dumping of dot matrix printers by the far east. Despite loud protests from printer manufacturers the EEC are also threatening similar tariff barriers against non dot matrix printers. At the same time several printer companies who produce their machines, in Europe, from imported components are being asked to prove that the majority of the assembly work is being done here and not in the far east.

1K UP AND STILL GROWING

Yes, not just 1000 members, but we have broken the 1K mark (1024). Before going to press with this issue this major landmark was reached and passed. What next I wonder?



YOUR LETTERS



Dear Editor,

May I ask your readers for some help? I am a carpenter and as such fit a lot of kitchens. I would dearly like to lay my hands on a design program of some sort, you know the sort of thing - (1) plot out room size with windows and doors - (2) Place pre-defined units onto plan and move around etc - (3) print out plan, with ideally a 3d projection. I am sure someone out there must use something like this and can let me have a copy for a price, or tell me where to get one. Can I also say that as a Spectrum 48k+ owner I am sad at the lack of serious software and magazines available for such a capable machine, I am thankful for FORMAT in bridging that gap.

Yours Sincerely, Raymond Hoy.

Dear Editor,

As there is a lot of effort that must go into the layout and printing of FORMAT, what about a disc magazine? That would save you a lot of trouble in printing and, after all, we are all disc users.

Yours Sincerely, Dr. Chin Lye Ch'ng.

Most people find reading from paper far easier than from a screen. Although a disc mag would be a lot better than the old tape based mags of a few years ago, with disc you could go back as well as forward. It would still be as hard to produce if it was to look good on screen. The main problem is disc formats, there is just no standard. Looking at membership forms shows 68% with 5.25" discs, 31% with 3.5" and 1% with 3". But then we need to look at the number of tracks, sides, and in very rare cases density. Add to this the cost of disc duplication and the result would be very expensive. However we are looking at ways to get more of the programs from FORMAT out on tape or disc. Ed.

Dear Editor,

In FORMAT#12 someone was after a keypad for his Spectrum 128. I've found the following firm is very helpful for most things SPECTRUM, so I suggest he contacts them direct. Combined Precision Components Ltd., 194-200 North Road, Preston, PR1 1YP. Tel: 0772-555034. I hope this is of use to your readers.

Yours Sincerely, R.T.Guy.

Dear Editor,

It occurred to me that it might be helpful if INDUG readers with mutual interests could be put in contact with one another. Why not invite members to agree to have their names and addresses published?

Yours Sincerely, L.Raybould.

That's just what the small ads section can now be used for. Remember that with over 1000 members, world wide, there must be someone who shares your interests. Ed.

Letters printed may sometimes be edited for length or clarity.

Greetings From Swansea

By: ALAN MILES.

Happy birthday INDUG - and well done, Bob Brenchley! I wonder just how many hours Bob has spent on the telephone answering people's questions, or at the computer producing FORMAT during this year? I know that whatever the figure is, it's been much longer than the normal working year - and the reason I know that is because of Bob's regular LATE-night calls to me, when he appears to be still in the middle of his working day.

All of us at MGT, Bruce, Jess, myself - and the new members of the team, Rob, Jill, Steve and Lynne - would like to thank Bob for the tireless and unremitting support he's given us over the last eighteen months or so. Without INDUG and FORMAT, it would have been impossible for us to provide the sort of support for our products that you customers quite rightly demand.

Here's to year 2, which promises to be tremendously exciting! Year 2 for INDUG and year 3 for MGT. The last year saw a number of important changes for us. First the move away from Rockfort Products to Cambridge, then the PLUS D and now Swansea and SAM - more about SAM later on.

Relocation to Swansea has been an extremely important move for MGT. Some of you may have read the report about us in ACE magazine where it described our Cambridge premises as having "not enough room to swing a soldering iron". In our view that was a generous exaggeration. With something like 500 orders being processed a month, we simply had no space to hold stock, nowhere to put people, and frankly, we just couldn't keep up. We looked hard in the Cambridge area for the room we needed to grow, but Cambridge prices have rocketed, and we'd have been spending all the money that we need to put into SAM on premises alone. So we looked elsewhere, and found that Swansea suited our needs perfectly.

Since June 1st, when we moved in, we've been able to handle our own production of PLUS D's and drives - and we're proud to report that of the 300 or so PLUS D's that we've already built and despatched from here, not one has reached the customer faulty, a problem that dogged us constantly when we sub-contracted production. We've also been able to take on a much wider range of products, as those of you who have received our recent customer price-list will have seen. And because we've been able to take on more people, we've been able to cut back into our shipping delays, which was concerning us above all. We're now projecting that we'll be able to ship the majority of orders within ten days by mid-August ... and we'd like to be better than that by the time SAM arrives.

Finally - a lot of you will have been waiting for this - we've finished the development of the long awaited TwoFace. In fact, I've seen the first working model today, the 5th of August, and we should be shipping the finished product by the week commencing August 22nd. (For those of you who don't know, the TwoFace is a two-way extension connector + joystick interface. A switch on the TwoFace allows you to turn on either or both of the interfaces you have connected. What this means is that previously incompatible interfaces will now work together - thus, Opus Discovery or Beta interface owners will be able to transfer their old files to the PLUS D or DISCiPLE ... and today I've seen two PLUS D's in harness on one Spectrum running four disc drives!).



Bruce Gordon (front) and Alan Miles.

But above all, the reason for the move to Swansea was to bring SAM to the market. All the changes I've already described are steps we felt we had to take if we were to make a success of SAM. We needed to have a small-scale production facility to make sure that the building of the early models was directly under our own control; we needed to have an expanded sales team which could give SAM customers the additional support they may require; we had to improve our response time to orders; and above all, we needed the peace and quiet which larger premises have given us, so that we had the right environment for creative thinking.

The release date for SAM is scheduled for February 1989. Earlier this year, we'd hoped that SAM would be out before Christmas, but the world-wide memory chip crisis made this an unrealistic target. At

first, it looked as though we would not be able to get the 256K x 4 memories that we require at all, but in the end we were able to negotiate a limited supply for the end of this year with a major manufacturer, with the promise of freer availability in the spring of 1989. This solved a problem ... and gave us a headache. If we launched before Christmas, we knew from the early response that we would be unlikely to satisfy demand. That would put us in the same position as so many other manufacturers before us - of long queues of customers waiting to receive product that they'd already paid for. Instead, we thought it would be better to delay the launch till a quieter time of year, when we could be more certain of component availability. There are other precautions that we're taking also, to make sure that we're not swamped with demand we can't meet. In the early

months, we've decided that in this country we'll sell mail order only - and that in the first place we'll offer SAM only to INDUG members or existing MGT customers. When we've seen what the demand is from this quarter, then we'll decide the quantities that we can make available for general release.

For this reason, we're asking INDUG members and customers to contact us at an early date if they're likely to want to buy. That'll help us to make sure that we're building and reserving the right numbers. If you haven't received an MGT information package by the end of August, and you're interested in getting details of how to make a reservation, please call us on 0792 791100 and we'll send you details. I must emphasise that if you make a reservation, this will not commit you to buy - and we don't want your money - yet! - but it will help us to organise ourselves ready for launch.

So what sort of computer is SAM going to be? Well, this is just to whet your appetite: - It's an 8-bit computer, not a 16-bit computer, but as Popular Computing Weekly says in its August 4th to 10th issue, the 8 bit computer is "Not dead - just resting". Certainly, SAM's features compare well with those of the 16-bit machines in circulation at the moment. - The basic computer comes equipped with 256K of memory, with a BASIC (based largely on BetaBasic which will make the entire memory available to programmers (unlike the Spectrum 128Ks with their banks of 16K memory)). - SAM will run the vast majority of your software and will be compatible with most existing hardware - including of course, the PLUS D and the DISCiPLE! - As well as a Spectrum-like screen mode there will be exciting new modes: SAM will have an 80-column screen capability, and we expect CP/M, Forth etc to be available from launch; another mode allows colour changes for every horizontal pixel, allowing high definition display. - On-screen colour will be programmable: as well as the 16 basic colours, a palette will allow you to select 1 from a range of 64 colours at any time. - Sam offers a host of on-board interfaces: for a mouse, a light-pen, a joystick, a MIDI, a DADI (a Domestic Appliance Digital Interface) and for a monitor and composite video as well as a TV. - The music chip is expected to provide 6 channel sound in stereo - we're experimenting with it at the moment. - SAM will have a professional-style keyboard with 8 function keys. - A built-in network will not only offer new opportunities in schools, but gives new scope for commercial software. We are encouraging software houses to write games in which one player/computer can be pitted against another player/computer, with the computer serving as assistant rather than opponent. That's all we can say now - but watch FORMAT closely for more news in coming issues.

Finally - I wonder if any of you FORMAT readers living in South Wales might be interested in looking at a position we'll be advertising soon - a sales person to help people on the phone and to take their orders. Qualifications? Willing to work long hours for little or no money, tactful, helpful, understanding, and with a total dedication to and understanding of the Spectrum. Bob It could be you ... At least I wouldn't get those late-night phone-calls.

Butting A Bit Less

By: Clyde Bish.

Hands up those who remember the 1k ZX81? I see other hands go up - in horror! These belong to the later computer fraternity who exclaim, "What on earth can you do with only 1k of memory!" The answer is quite a lot - IF you know how to make the most of the memory available. (Someone even wrote a chess player in 1k).

But surely we don't have that problem now. After all, you must have at least 48k on board. Yes - and No. The 48k Spectrum has only about 41k of usable memory and this gets used up very fast in adventures, DATA bases and other elaborate programs. All is not easy in the 128k world either. The processor can only address 48k at a time and this can be quite a nuisance when you've almost got to the end of a program and you run out of space. What can you do? Page in extra memory - if you can or - Byte a Bit less! Read on.

Lets make a start with those great byte-wasters, numbers. You might think that to hold the number .134 in memory would take up just three bytes, one for each digit, but it doesn't. Type in this short program then press ENTER:

```
10 LET a=.134: LET p=PEEK 23635+256*PEEK 23636: FOR f=p TO p+17:
PRINT f, PEEK f: NEXT f
```

Now RUN, and you'll see displayed two columns of numbers. The left hand one shows the memory spaces you have used. Opposite each of these is a number which is the code of the character held in each of these bytes. The first eleven of these are the line number and codes for the beginning of your program - LET a=.134. You'll see that 6 other numbers follow the code for your 4 (52) before you come to the 58, which is the code for the colon you typed in. The first of these is 14. This is the code for the number character - see Appendix A in your manual - and is followed by five other numbers. These represent your number .134 in binary. This is the number system your computer actually uses when it runs your program. (It only displays numbers in denary for the sake of us 10-digit humans!).

What this amounts to is that storing numbers in a program takes up six more memory spaces than it would first appear. (In the above example 10 instead of 4.).

So, numbers are great byte-wasters. But who needs them anyway? Here are some tricks which avoid the use of numbers. Each example is followed by a bracket showing (number of bytes used/number that would be used if any actual number had been included).

The numbers 0, 1 and 3 can be replaced by functions. So zero can be replaced by NOT PI (2/7) because NOT any value equals zero (and the value of PI, 3.142 etc is built into the machine's ROM). In a similar way the value 1 can be replaced by SGN PI (2/7) as the SIGN of any positive value equals 1, and 3 can be replaced by INT PI (2/7) as the INTEger of a number removes any digits after the decimal.

Other numbers can be produced in a variety of ways. If you have already declared a variable, for example by having a statement such as

```
10 LET L= SGN PI
```

you can use this value, and subsequent ones to declare others. Look at this example which saves 8 bytes.

```
20 LET z=L+L      in place of 20 LET z=2
30 LET f=z+z      (22)         30 LET f=4 (30)
```

VAL, the function which eVALuates a string, can also be used. For example 10 LET z=VAL "2" (12/15) again makes 'z' equal to 2. The VAL trick can be used for any number but there are other tricks which can be used for certain numbers which save even more bytes. For numbers from 32 to 255 the CODE function can be made use of. For example 10 LET s=CODE " " makes 's' equal to 32, the CODE of the character space. (Look at Appendix A for the other values). Similarly 20 LET x=CODE " COPY " makes 'x' equal to 255, but there is a problem. COPY is a keyword, but the quotes after CODE leaves the L cursor, and you need a K. This is how you need to enter that line (ignore the / signs - they are just there as spacers) 2/0/LET/x/= /CODE (in E Mode)"/THEN/COPY/" Now cursor left, delete the THEN, and there you are!

One final trick for large numbers is to use VAL but with the E-form of the number. So 10 LET t= VAL "1E6" makes 't' equal to 1000000 (i.e. 1 Extended by 6 zeros) and takes up 6 bytes less than VAL "1000000" (and 7 bytes less than simply using the number value 1000000).

It may not seem that you're saving much. After all, what's a few bytes among 41 thousand odd? The answer is a lot, when you add them all up! Remember, "Look after the bytes, and the K's will look after themselves"! A colleague, writing a number crunching spread sheet-type program used these tricks and saved 2.5K on a 12K listing!

But this isn't the end of the story. (It's really just about the beginning!) You can save even more space with numbers (and other types of variables) if you don't intend to change them while the program is running, (or don't mind if they are changed). When you RUN a program, any variables you set are added to what is known as the variables area. This makes them immediately available for the program to use. This means that you have them stored twice, once in the program area, and again in the variables. What a waste of space! Why not just store them in the variables area (known as VARS for short)? Try this. Type

in LET a=10 with no line number, and press ENTER. "O.K." says the computer now type in PRINT a and press ENTER. The number 10 appears on screen. Although you don't hold it within your program - you haven't got a program! - the machine knows the value of 'a' because its in VARS. (You can prove this by typing in a program similar to the one given earlier, replacing the PEEK values with 23627/8 and missing out the LET a=134 - the machine already knows what a is! Start your program with GOTO 1, NOT RUN - I'll explain why in a moment. You should be able to find the code for 'a', followed by the representation of 10 in five digits).

So, why must you not use RUN with programs where the variables are only held in VARS? Quite simply, RUN clears the VARS area, and so goodbye to your information! The golden rule is SAVE such a program with a line number so that it autostarts, (and remember to use GOTO if you BREAK).

Finally for this month, declare a variable for a number if you intend to use that number more than twice in a program, and remember that single letters are much less wasteful than variable names. You can choose ones which look like the numbers they will represent. I always use 'i' for 1, 'z' for 2, 'o' for 0 and 'p' for 3 (it reminds me of PI)

ADDRESS AND DISC MANAGER

For PLUS D and DISCiPLE (version 3 onwards)

THREE programs, on one 3½" disc, handle all your Address and Disc organisation at a truly realistic costing.

007 MENU. With the enormous number of K per disc, this program is essential to keep a track of which disc your program(s) is on. Simply insert your disc(s) then press a key and a Full CAT is held in a record (Upto 2200 Records). Can SEARCH for any program and INSTANTLY tell you which disc its on (and even LOAD it).

007 SUPERFILE. Holds upto 500 name/address records and finds any one INSTANTLY. Super Fast SEARCH, ALPHA SORTs, etc. Prints as FILES or LABELS by SEARCH, Sort or String.

007 LISTFILE. Holds 1,000 single line records. Fast SEARCH, SORT, ALPHA SORT, RENUMBER. Can print all or selected records.

ALL programs can Print Out to ANY type of printer.

All three programs on one 3½" disc.... **£9.95**

ZX-GUARANTEED (G.A.Bobker) Dept: INDUG. 29 Chadderton Drive, Unsworth, Bury, Lancs, BL9 8NL. Tel: 061-766 5712 (Do NOT phone if STAR TREK is on TV)

THE HELP PAGE

Problems with your DISCIPLE, PLUS D or Spectrum. Dont worry, let the HELP PAGE sort them out. Note: One question per letter please.

T.P.I.

I use the 3.5" disc drive (DSDD) as supplied by MGT. Is it possible to use 135tpi discs? Also what is tpi?

R.Hodge. Camberley.

This is a very common question, tpi confuses many people new to disc drives. It stands for TRACKS PER INCH, there are 135 tracks per inch on a 3.5" disc designed for 80 track drives. On a 5.25" disc this would be 96tpi, as you can see 3.5" discs pack their data into a smaller space than their bigger brothers. So for the 80 track drive you bought from MGT you should use 135tpi discs for real security, but you wont damage your drive using lower tpi discs, you just might lose programs though.

TASPRINT

I recently purchased TASPRINT for use with Tasword 2. However, the program is not much help with configuring it to work with the PLUS D, which was not around when it was written. How do I get it working?

Alan Gibbons. Derby.

Sorry Alan but TASPRINT wont work with the DISCIPLE or PLUS D. It uses the printer channel (#3) in a strange way, and, if you said yes to a printer in your system file, your interface takes control and TASPRINT falls down. However QUALITAS, from Seven Stars Software, does the same sort of job so give that a try.

CODE CRASH

I am trying to write a small machine code routine into my Spectrum +2. The routine works in 48k mode but always crashes in 128k mode. What am I doing wrong?

S.Fowler. Manchester.

According to the listing you sent me your program POKES in the machine code above 60000, you then do a NEW and load in a program which uses the routine. Fine, machine code has been entered this way since the very first Spectrum rolled off the production line. But then along came Amstrad... I've said it before and no doubt will do again, the +2 has BUGS. In this case a BIG BUG, for some unknown reason some locations above RAMTOP are corrupted when you do a NEW in 128k mode. The old Sinclair 128k didn't suffer from this so its another fault down to Sugar's boys, of course they deny the problem exists but then they deny everything. Try POKEing 255 to all locations above 60000, type NEW, then PEEK them. Locations 65316/7 get corrupted and sometimes others.

More from the HELP PAGE next month..

ME & MY COMPUTER

ONE PERSONS USE OF THE SPECTRUM

By: C.Grant DIXON.

My interest in electronics started when I built a one-valve receiver at the age of 11, that was in 1927. I later built a 30-line television receiver when I was in the sixth form, and I have had an abiding interest in television since then. In 1959 I heard about a system of television using a slow scanning speed, in which the video signal consisted of audio tones which could be recorded on a normal cassette recorder. I have been an enthusiastic Slow-Scan TeleVision (SSTV) supporter ever since. In the mid-1970s a friend who saw my equipment said casually "you should keep your eye on these microprocessor chips, they should be quite useful in this work". So it was then I was introduced to the computer as a controlling instrument...NOT as a device for playing games.

The first computer that I built was the 8080-based TRITON featured in E.T.I. magazine in 1978/79. I soon had this linked up to my scan-converter and in Jan 1982 I was storing SSTV pictures in the computer memory and printing them out with an Epson printer. Later I built a TANGERINE computer (also called a MICROTAN) which was 6502 based. With various extensions added to it I now had the facility for storing on discs but I found that programming the 6502 chip did not come so easily to my ageing brain as the 8080 code had done.

In May 1987 I treated myself to a commercial computer...I bought a second-hand 48k SPECTRUM, rubber keys and all. I thought that as there were a lot of Spectrums about, I might find some interesting programs. I therefore went out and bought one copy of each of the magazines devoted to Sinclair computers and was completely taken aback by the large volume of colourful games; this was NOT what I was looking for!

By this time I had also become interested in weather satellite reception so I got started and wrote a program which would put a map of the northern hemisphere on the screen and then draw the satellite's track for any value of longitude of the equatorial crossing. But Europe seemed very small on this map so I wrote another program which displayed a map of Europe and plotted the satellite's position in real time.

My next project was a re-write of the TRITON program which would print out SSTV pictures, the main advantage now being that I had a disc system to store the pictures. This involved linking the scan-converter to the Spectrum and a small interface board was constructed carrying a Z80PIO and some decoding logic to give one input port and one output port. This board was plugged into the back of the Disciple and has proved to be an extremely useful addition. (Thank heaven I got a Disciple and not a

Plus D...that extra socket is really needed.) Line pulses, Frame pulses and 4-bit video are fed into the Spectrum's input port and timing is done by polling the pulses and using a delay routine between picture elements. Once a frame has been loaded into the Spectrum it may be displayed by feeding the contents of the Spectrum's memory back to the scan-converter whence it is viewed by the SSTV monitor in the normal way. The computer has substituted the picture in its own memory for the normal camera picture.

The Spectrum's screen is not so useful for displaying these pictures as it is not possible to get 16 grey levels at each pixel. To get 16 levels one can use a little square of 4 x 4 dots in which 0 to 16 dots can be printed and this square can be used to represent a pixel. This works well with the printer and we can get 128 x 128 pixels printed without needing to purchase a wide-carriage printer. With the screen, however, we have 192 lines of 256 dots available and this gives us only 48 x 64 SSTV pixels out of the 128 x 128 in the whole picture; hence the need to rely on the original display device.

I suddenly realised that, if I could print SSTV pictures, then I could equally well do the same thing with the digitised signal in the weather satellite receiver. The result was a similar program which enables me to print out weather pictures and store them on disc. The actual weather picture as seen on the TV screen is 256 x 256 pixels but we are only able to deal with half this resolution, even so one does get a good general impression of the cloud cover.

The most recent project is a re-write of a program which I wrote for the Microtan a couple of years ago to allow circuit diagrams to be drawn on the screen, stored to disc and printed out as required. This new Spectrum program is actually much improved and easier to use, the only snag being the size of the lettering available which limits the amount of textual annotation one can add to the diagram. It is fairly easy to draw large diagrams by drawing on several screens and using a cut-and-paste technique.

I have been most impressed by the Disciple interface which turns the Spectrum into a 'real' computer. The speed of the DOS is remarkable, being 3 or 4 times faster than the DOS I was using on the Microtan. Even the cassette loading is fast when compared with the 1200Baud of my Triton.

Well, there is my personal use of the Spectrum + Disciple combination; if there are any FORMAT readers who share a similar interest I should be glad to hear from them.

C. Grant Dixon,
Kyrle's Cross,
Peterstow,
Ross-on-Wye,
HR9 6LD.

Do you use your computer for something a little unusual? Tell us about it, you may start others thinking.

SIDE-WRITER

By: Dave Kennedy.

Here's a small utility I think you'll like, it solves what for me was a major problem, addressing envelopes on my dot-matrix printer. Most printers won't feed an envelope reliably unless its quite large, you end up getting the last few lines printed on to of each other.

SIDE-WRITER gets round the problem by letting you feed the envelope through long-ways, and then printing the address on its side - hence the name. It works with any printer with bit image graphics, just change the codes around lines 710 & 1190 to match. As set it works with most Epson compatibles. The routine is fully relocatable. To use it just load it in, RANDOMIZE USR the load address then type in the address you want to print. When finished press STOP (symbol-shift A). EDIT will abort the routine and return you to basic. Experiment a bit, you'll soon get the hang of lining things up as you want.

```

10      ;SIDEWRITER. By Dave Kennedy.
20      ORG 50000
30 SIDEWR PUSH BC      ;"BC" = LOAD ADDRESS
40      POP IX          ;"IX" IS USED AS CODE LOCATION CONSTANT
50      LD HL,DATASTART-SIDEWR ;RELATIVE START OF DATA FIELD
ELD
60      ADD HL,BC      ;REAL START OF DATA FIELD
70      LD (IX-1),H
80      LD (IX-2),L
90      PUSH HL        ;DATA START
100     LD A,2
110     CALL #1601      ;SELECT UPPER SCREEN
120     LD A,13
130     RST 16
140     POP BC          ;DATA START
150     LD H,B
160     LD L,C          ;NOW "HL" = "BC" = DATA START
170 INPUT PUSH BC      ;PRESERVE DATA START
180     LD A,">"
190     RST 16
200     LD A,8
210     RST 16          ;BACKSPACE OVER ">"
220     PUSH HL         ;CURRENT DATA POSITION IN BUFFER
230 INPUT0 CALL #10AB   ;READ KEYBOARD
240     JR C,CHARIN     ;& EXIT IF KEY PRESSED
250     RES 3,(IX+2)    ;TO PREVENT PRINING ON LOWER SCREEN
260     JR INPUT0       ;REPEAT UNTIL KEYPRESS
270 CHARIN POP HL      ;CURRENT DATA BUFFER POSITION
280     POP BC          ;START OF DATA BUFFER
290     CP 7            ;EDIT KEY?
300     RET Z           ;STOP INPUT & EXIT TO BASIC

```



```

310 CP 12 ;DELETE KEY?
320 JR NZ,INPUT1
330 AND A
340 SBC HL,BC ;CURRENT - START IN DATA BUFFER
350 ADD HL,BC
360 JR Z,INPUT ;ALREADY AT BUFFER START SO NO DELETE
370 LD A,8
380 RST 16 ;BACKSPACE
390 LD A," "
400 RST 16
410 LD A," " ;OVERWRITE PREVIOUS CHAR. & ">"
420 RST 16
430 LD A,8
440 RST 16 ;THEN BACKSPACE
450 LD A,8
460 RST 16 ;BACKSPACE
470 DEC HL ;REDUCE CURRENT DATA BUFFER POSITION
480 JR INPUT
490 INPUT1 LD (HL),A ;SAVE KEY VALUE INTO DATA BUFFER
500 CP 13
510 JR NZ,INPUT2
520 LD A," " ;OVERWRITE ">" IF AT LINE END
530 RST 16
540 LD A,13
550 INPUT2 RST 16 ;PRINT KEY VALUE
560 LD A,(HL) ;RECOVER KEY VALUE
570 INC HL
580 CP 226 ;STOP KEY?
590 JR NZ,INPUT
600 LD (HL),A ;PLACE "STOP" AT END OF DATA BUFFER
610 DEC HL
620 LD (HL),13 ;N/LINE
630 INC HL
640 INC HL ;END OF DATA BUFFER + 1
650 LD (IX-3),H
660 LD (IX-4),L
670 LD A,3
680 CALL #1601 ;SELECT PRINTER
690 LD A,27
700 RST 16
710 LD A,48 ;SET TO 8 LINES/INCH = CHARS./INCH
720 LD B,A ;SET BC TO HIGH VALUE FOR "CPIR"
730 RST 16 ;SET TO 8 LINES PER INCH
740 LD H,(IX-1)
750 LD L,(IX-2) ;"HL" = START OF INPUT BUFFER
760 LD (IX-5),0 ;ZERO LINE COUNTER
770 FIND LD A,13
780 PUSH HL ;SAVE CURRENT START OF LINE
790 CPIR ;& FIND NEXT LINE END
800 INC (IX-5) ;LINE NUMBER COUNTER
810 LD A,(HL)
820 CP 226 ;STOP KEY
830 JR NZ,FIND ;STILL NOT AT END OF DATA BUFFER
840 LD H,(IX-3)
850 LD L,(IX-4) ;START OF SPARE MEMORY
860 LD B,(IX-5) ;NUMBER OF LINES IN INPUT BUFFER
870 SETUP POP DE ;RECOVER START OF EACH INPUT LINE
880 LD (HL),E

```

```

890 INC HL ;& STORE EACH VALUE IN REVERSE ORDER
900 LD (HL),D
910 INC HL ;EVENTUALLY START LINE IS STORED AT END
920 DJNZ SETUP ;REPEAT FOR EACH LINE
930 NEXT LD (IX-6),B ;ZERO PRINTED LINE COUNTER
940 LD B,(IX-5) ;NUMBER OF LINES IN INPUT BUFFER
950 LD H,(IX-3)
960 LD L,(IX-4) ;NEXT CHARACTER TO PRINT IN EACH LINE
970 NXTLIN LD E,(HL)
980 INC HL
990 LD D,(HL)
1000 LD A,(DE) ;BUFFER CHARACTER TO PRINT
1010 CP 13
1020 JR NZ,PRINT ;STILL NOT AT END OF LINE
1030 INC (IX-6) ;ANOTHER LINE END FOUND
1040 LD A,32 ;REPLACE NEWLINE WITH A SPACE
1050 DEC DE ;& PREVENT CONTINUING INTO NEXT DATA LINE
1060 PRINT INC DE ;NEXT CHARACTER IN LINE TO PRINT
1070 LD (HL),D
1080 DEC HL
1090 LD (HL),E ;& STORE IN LINE POSITION BUFFER
1100 INC HL
1110 INC HL ;ONTO NEXT LINE ADDRESS
1120 PUSH HL ;LINE ADDRESS STORE
1130 PUSH BC ;LINE COUNTER
1140 GRAPHI LD L,A ;KEY VALUE
1150 INC L
1160 LD H,0 ;"HL" = ASCII KEY VALUE +1
1170 LD A,27
1180 RST 16
1190 LD A,76
1200 RST 16 ;SET PRINTER TO DOUBLE DENSITY MODE
1210 LD A,16
1220 RST 16
1230 XOR A
1240 RST 16 ;& 16 COLUMNS OF DATA TO FOLLOW
1250 LD B,8 ;8 ROWS TO PRINT FOR EACH CHARACTER
1260 LD DE,(23606) ;SCREEN CHARACTER START - 256
1270 ADD HL,HL ;X2
1280 ADD HL,HL ;X4
1290 ADD HL,HL ;X8
1300 ADD HL,DE ;NOW AT DATA FOR NEXT ASCII KEY
1310 ROW DEC HL ;END OF DATA FOR PRINT CHARACTER
1320 LD A,(HL)
1330 RST 16
1340 LD A,(HL)
1350 RST 16 ;PRINT TWICE
1360 DJNZ ROW ;& REPEAT FOR ALL 8 ROWS
1370 LD A," "
1380 RST 16
1390 POP BC ;DATA LINE COUNTER
1400 POP HL ;DATA ADDRESS BUFFER
1410 DJNZ NXTLIN ;REPEAT FOR ALL DATA LINES
1420 LD A,13
1430 RST 16 ;ELSE PRINT NEWLINE
1440 LD A,(IX-5) ;NUMBER OF LINES IN DATA BUFFER
1450 CP (IX-6) ;NUMBER OF LINES CURRENTLY PRINTED
1460 JR NZ,NEXT ;STILL NOT AT LONGEST LINE END

```

```

1470      RET          ;ALL LINES PRINTED
1480
1490 DATA DEF B 0      ;PLACE INPUT DATA BUFFER HERE
1500      ;AND FOLLOW WITH LINE ADDRESS STORE

```

For those of you without an assembler I've included this basic poke program, remember you can load and run at any address.

```

10 FOR I=0 TO 242: READ A: POKE I+50000,A: NEXT I
20 DATA 197,221,225,33,243,0,9,221,116,255,221,117,254,229,62
30 DATA 2,205,1,22,62,13,215,193,96,105,197,62,62,215,62,8,215
40 DATA 229,205,171,16,56,6,253,203,2,158,24,245,225,193,254
50 DATA 7,200,254,12,32,24,167,237,66,9,40,222,62,8,215,62,32
60 DATA 215,62,32,215,62,8,215,62,8,215,43,24,204,119,254,13
70 DATA 32,5,62,32,215,62,13,215,126,35,254,226,32,187,119,43
80 DATA 54,13,35,35,221,116,253,221,117,252,62,3,205,1,22,62
90 DATA 27,215,62,48,215,6,255,221,102,255,221,110,254,221,54
100 DATA 251,0,62,13,229,237,177,221,52,251,126,254,226,32,243
110 DATA 221,102,253,221,110,252,221,70,251,209,115,35,114,35
120 DATA 16,249,221,112,250,221,70,251,221,102,253,221,110,252
130 DATA 94,35,86,26,254,13,32,6,221,52,250,62,32,27,19,114,43
140 DATA 115,35,35,229,197,111,44,38,0,62,27,215,62,76,215,62
150 DATA 16,215,175,215,6,8,237,91,54,92,41,41,41,25,43,126
160 DATA 215,126,215,16,249,62,32,215,193,225,16,195,62,13,215
170 DATA 221,126,251,221,190,250,32,172,201
200 SAVE D1"SIDEWRT_C" CODE 50000,243

```

Bradway Software

Letta-Head Plus

Still the most versatile Spectrum utility to design and print your own business & personal stationery: letterheads, receipts, orders, labels, posters etc. Create the design on screen, select the required format & print all the copies you need.

- Library of 25 fonts including foreign alphabets.
- Fast, compiled editor to modify & create new fonts.
- All characters proportionally spaced.
- Choice of 81 font sizes.
- Screen windows may be inverted, scrolled & erased.
- Enclose any area in a box of any thickness.
- Graphics draw, arc, circle, plot, fill for logo design.
- Load and use SCREENS from any other program.
- Print Headings in a choice of 18 sizes.
- Design labels up to 4" x 2" in size.
- Single key selects cassette label option.
- Price £9.50 (cass) £10.50 (mov, disc).

Dumpy

All the screen dumps you will ever need for your Spectrum! Dumpy is a unique screen dump generator, from a list of your requirements it creates the machine code, relocates it and saves it ready for you to use in your programs. No need to understand assembler, just follow the menu.

- Define starting & ending line & column.
- Optional automatic display area determination.
- Handles full 24 lines of the display.
- Select from up to 7 print details.
- Select from 9 widths and 9 heights of output.
- Plain black & white or shaded colour scale.
- Tab to any position on the paper.
- Drives any width printer.
- Machine code can be positioned anywhere in RAM.
- Portrait or landscape dumps (for big posters).
- Detailed manual with examples for m/code novices.
- Price £9.00 (cass) £10.00 (mov, disc).

Letta-Head & Dumpy require an Epson compatible printer. All Bradway Software programs drive almost any printer interface (including Disciple & Plus D) and are available on 5.25" or 3.5" disc for Discoversy or Disciple. Post & Packing: UK & Europe included, please add £1.50 per program world-wide airmail. Payment by cheque, PO, GIRO 65 675 0901, ACCESS. Send for our full catalogue of utility programs for the Spectrum.

"Hillsett", Upper Padley, Grindleford, Sheffield, S30 1JA. phone (0433) 30799.

REVIEW DUMPY REVIEW

By: John Wase.

Dumpy, from Bradway Software, is a screen-dump generator. O.K, I know the DISCIPLE and PLUS D have their own screen dump routines, but the big advantage of Dumpy is the vast choice you have.

You can choose landscape (printed normally across the paper) or portrait (down the paper). You can select from a whole range of heights and widths. You can choose any section of the screen (including the bottom two lines) and output a dump to start at any printer column. You can even choose autosize (prints out only the area of the screen with anything on it). You are offered around 20 different printer interfaces, so you are not just stuck with your Disciple: you can stick the lot on tape and go back to your old Kempston interface in emergencies.

Dumpy appears to deal with most eventualities. The most common queries I get concern printers and their interfacing - Dumpy comes with special built-in software for a whole range of printers, with a choice of plain or shaded output, and with a variety of print densities with provision for early printers and the latest types.

All right then, you can get any output you want for virtually any dot-matrix printer. So what? Well, as I said Dumpy is a screen-dump generator. It generates code to fit in any location (including the 48K printer-buffer). It gives you the chance to test and save it before use. And you can call it with a simple RANDOMISE USR command. Pretty good. The program is well error trapped, and there is a very clear manual, complete with a help-line for those who get stuck.

I've used Dumpy for a number of years and found it very useful, particularly for program conversions to disc and full-size printer (replacing the COPY command...). Perhaps familiarity breeds contempt, but the disc version wouldn't work at first - until I remembered to do a POKE @6,1 so the printer control codes could get through. I tested it with an FX80 printer (coward) and both PLUS D and Disciple 3b: it worked splendidly after the poke.

If I have a reservation, it is the shading pattern: one of lines rather than absolute shading; but that's a small quibble compared with the overall usefulness of the package.

The program is available from BRADWAY SOFTWARE, Hillsett, Upper Padley, Grindleford, Derbyshire, S30 1JA, (note their new address) and represents excellent value at £9.00 (cassette), £10.0 (disc) including UK postage.

Fun Glossary

A computer glossary that just has to taken with a pinch of salt.

- 1) **COMPUTER:** A machine which does exactly as its told, without exception, even when you meant to tell it to do something else.
- 2) **PROGRAM:** A set of instructions telling a computer how to loop and halt unexpectedly.
- 3) **STORAGE (1):** The disc (tape, used envelope etc) you store your latest program on and then cant find because you didnt file it in its right place. **(2):** The box (drawer, cabinet etc) that you always meant to put that missing disc in.
- 5) **HIGH LEVEL LANGUAGE:** A programming language in which instructions are written in a form of English which requires a degree in computing to understand. e.g BASIC.
- 6) **DEBUGGING:** The process of replacing one error with another.
- 7) **WORKING PROGRAM:** A program from which all minor errors have been removed.
- 8) **COMPILER:** A program to convert your errors into machine code.
- 11) **BACKUP COPY:** Something you do not have time to do because of all the disc crashes.
- 12) **FLOWCHART:** A pictorial representation of logic errors.
- 15) **ENHANCEMENT:** A way of introducing new errors into a program.
- 16) **SORT:** A program, that takes hours to put your data in order, so you save .03 seconds each time you do a search.
- 18) **PUNCHED CARD:** The most efficient form of data storage ever invented. The data is held as a hole in the card. As the card only serves to hold the holes in place, it's 100% redundant.
- 20) **FLOPPY DISC:** The same as a punched card but with one big hole in the middle and a smaller one slightly offset from the first.
- 22) **SPECTRUM:** The only computer you can't poke fun at, because its the best machine in the world...

If you have any other fun definitions lets have them for a future issue.

HACK-ZONE

By: Hugh J. McLenaghan.

Here we are again, another month has passed. Last month I promised a KERNOV but this has proved bigger than I first thought, I will bring you more news next month. The first thing this issue is a routine to load in sections of a 128K Snapshot file so that you can Hack them. You do not need a 128K spectrum to hack a 128K game, as only 16K will be loaded in at once.

128K SNAPSHOT HACKER.

```

1 REM 0123456789012345678901234567890123456789012345678901234
567890123456789012345678901234567890
10 REM 128K Snapshot Hacker.
20 REM Written by
30 REM Hugh McLenaghan
40 REM on 12/6/1988.
50 REM
60 CLEAR 32000: GOSUB 9000: CLS
70 PRINT "Insert the Games Disc then pressENTER."
80 IF INKEY$ <> CHR$ 13 THEN GOTO 80
90 IF INKEY$ = CHR$ 13 THEN GOTO 90
100 CAT *!: REM Shortened version of the catalogue.
110 DIM n$(10)
120 INPUT "Name of 128K Snapshot file ?:"; LINE n$
130 IF n$ = " " THEN GOTO 120
140 FOR A=1 TO 10: POKE 23296+A, CODE n$(A): NEXT A
150 RANDOMIZE USR 23760
155 IF PEEK 23296=255 THEN PRINT N$;" not on this Disc."" Pres
s any key.": PAUSE 0: RUN
160 IF PEEK 23296 <> 9 THEN PRINT "Not a 128K Snapshot file!"" P
ress any key.": PAUSE 0: RUN
170 CLS
180 PRINT "Found ";n$
190 PRINT "Which page do you wish to load (0-7)?"
200 LET A$=INKEY$: IF A$ < "0" OR A$ > "7" THEN GOTO 200
210 PRINT A$
220 PRINT "SURE (Y/N)?"
230 POKE 23658,8
240 LET Z$=INKEY$: IF Z$ < "Y" AND Z$ < "N" THEN GOTO 240
250 POKE 23658,0
260 IF Z$ = "N" THEN GOTO 170
270 LET PAGE=VAL A$
275 LET BYTE=(16384*PAGE)+1: REM Displacement of page within 12
8K Snapshot file.
280 LET SECTORS=INT (BYTE/510): REM Number of sectors which hav
e to be loaded before the page required is found.
290 LET BYTES=BYTE-(SECTORS*510): REM Displacement in bytes of
the first byte of the required page within the first sector.
295 LET ADD=32768-BYTES: REM Address where the sector has to lo

```


EXPANDING

TASWORD 2

By: Ken Elston.

Tasword 2 is probably the most widely used wordprocessor in the history of home computing. I got my first copy in early 1983 and used it with tapes, as that was the only means of storage available at that time. When microdrives came out in late '83 I upgraded to the microdrive version and marvelled at the extra speed of storage.

However, familiarity breeds contempt (or so they say) and after awhile I felt the need for a few extras, those little features that other word processors had but TW2 lacked. Well I now have some 12 different versions of TW2, each set up to do a special job, each giving me a feature (or 2 or more) that I needed to cope with a particular application. Just to give you some idea of the extra features look at this list:- Screen Colour Change, Word Count, Headers, Page Numbering, Multi Copies, 128 Characters per line, Printer Spool Files, Larger text file (350) lines, Transmission over RS232, plus many more.

I'm now trying to bring many of these features together in one version, and I thought FORMAT readers would like to join me on this Tasword Expanding Journey. This short article looks at some of the key addresses in TW2 and helps you customize the screen colours to your own taste. In future articles, which will appear every now and then, I will cover as many of the new features as I can fit in to TW2.

So lets start off with a list of addresses within the CODE file of Tasword 2. Its a very messy program, but over the years I've managed to track down most of the important locations so here goes:-

54784	Start of machine code.
54784	Help Pages.
58053	Ascii table codes 32-126
59000	Messages
59081	Entry point to clear file on loading.
59153	Key click routine.
59232	Routine to print 2 characters to screen.
60928	Graphics Table (for 32col mode)
61184	Graphics table (64 col characters)
62216/7	Pointer to start of file.
62279	Return to basic.
62472/3	Vector to print routine.
62636	Control table (symbol shifted characters).
62710	Control table (Extended mode).
64150/1	Length of file minus 22.
64330	Main entry point.
64333	Rentry point (after Search/Replace)

aded to make sure that the first byte of the required page is put into 32768.

```
300 CLS
310 PRINT AT 10,7; FLASH 1;"Loading Please Wait."
320 LET T=PEEK 23309: REM Get the TRACK number from the directory sector.
```

```
325 LET S=PEEK 23310: REM Get the SECTOR number from the directory sector.
```

```
330 IF SECTORS=0 THEN GOTO 400
340 LOAD @*,T,S,ADD: REM Load the required sector.
350 LET T=PEEK (ADD+510): REM Get next TRACK number.
360 LET S=PEEK (ADD+511): REM Get next SECTOR number.
370 LET SECTORS=SECTORS-1: REM Reduce SECTORS remaining.
380 GOTO 330: REM Repeat until SECTORS=0.
390 REM LOAD DATA.
```

```
400 LET SECTORS=INT (1+(16384/510)): REM Sectors to be loaded so that AT LEAST 16384 bytes are loaded.
```

```
405 LOAD @*,T,S,ADD
410 LET T=PEEK (ADD+510)
420 LET S=PEEK (ADD+511)
430 LET ADD=ADD+510: REM Increase Destination address of code.
440 LET SECTORS=SECTORS-1
450 IF SECTORS<>0 THEN GOTO 405
```

```
460 CLS
470 PRINT "Page ";a$;" loaded into 32768-49151."
```

```
480 INPUT "Save as what?"; LINE N$
490 IF N$="" THEN GOTO 480
```

```
500 PRINT "Insert Disc, Then press any key."
```

```
510 IF INKEY$="" THEN GOTO 510
```

```
520 SAVE D*;N$ CODE 32768,16384
```

```
530 PRINT "Saved. Load another Page (Y/N)?"
```

```
540 POKE 23658,8: REM Turn CAPS LOCK on.
```

```
550 LET A$=INKEY$: IF A$<>"Y" AND A$<>"N" THEN GOTO 550
```

```
560 POKE 23658,0: REM Turn CAPS LOCK off.
```

```
570 IF A$="N" THEN RANDOMIZE USR 0
```

```
580 RUN
```

```
9000 REM Set up Machine-Code.
```

```
9010 IF PEEK 23760=17 THEN RETURN
```

```
9020 LET A=23760
```

```
9030 READ Z: IF Z<>999 THEN POKE A,Z: LET A=A+1: GOTO 9030
```

```
9040 RETURN
```

```
9050 DATA 17,1,0,33,1,91,243,213,229,207,63,219,187,33,214,27
```

```
9060 DATA 209,213,229,35,6,10,26,190,32,6,35,19,16,248,24,18
```

```
9070 DATA 209,209,213,33,214,28,229,35,6,10,26,190,32,19,35,19
```

```
9080 DATA 16,248,225,17,0,91,1,0,1,237,176,211,187,225,209,251
```

```
9090 DATA 201,225,225,209,211,187,28,123,254,11,32,187,30,1
```

```
9100 DATA 20,122,254,4,32,179,62,255,50,0,91,251,201,999
```

I will now explain some things about the routine. The name of the program which you want to search for is put into 23297-23306. After the machine-code is called it searches the directory of the disc until the program is found. If it cannot find the program, then it POKES 23296 with 255. If the program is found, then the file's information is loaded into the printer buffer (only the 256 appropriate bytes are moved here). The rest of the program does the hard work of loading the data.

See you next month.

64336 Main get key loop
64704 Control table (Cap-Shifted characters).

Well thats just some of them. Now lets look at screen colours, TW2 uses the standard Spectrum - white paper, black ink - colour system which works, but could be better. When I write a program I prefer blue paper and white ink, its more restful on the eyes and makes small characters easier to read so this was one of the first I set out to change in TW2. The colours are set in TW2 by bit manipulation which makes them rather difficult to alter but it turned out to be easy to alter the instructions to LD (HL),N - CODEs 54,N - with N being the colour, so thats the course we will take.

Starting with the main Paper/Ink for Taswords 64 column display. Like most Spectrum colours this will be stored as a single number calculated as INK + PAPER * 8 (lets forget FLASH and BRIGHT shall we). So to set blue paper and white ink the calculation is $7+1*8$, which your computer will soon tell you gives a result of 15. OK, with me so far, well I promise thats the maximum amount of maths this article will require. Now main screen colours get set at two locations so load in TW2, exit to basic and enter the following four pokes:

POKE 58512,54
POKE 58513,15
POKE 58521,54
POKE 58522,15

Now if you RUN the program you will have a blue screen with a white cursor flashing in the top left hand corner. Right but we still have a white border and the status lines (lines 23 & 24) are still black and white. Border is no problem, it needs just one poke and thats the straight colour number 0-7, so POKE 64616,1 to set it to blue. TW2 does not restore the border colour on return to basic although it restores all other colours. The lower two lines are just a little more difficult. I set them to Red/White, the poke for line 23 is calculated as above ($7+2*8 = 23$) so POKE 64570,23. The last line is printer INVERSE so we need to calculate the colour as $2+7*8$ which equals 58. So POKE 59993,58 completes the change

Next we come to the Margin colours. To reset these (I use Cyan paper/Black ink) enter the following four pokes:

POKE 58508,54 - Left Margin.
POKE 58509,40
POKE 58517,54 - Right Margin.
POKE 58518,40

Finally there's 32 column mode, not used by many I know, But I wont ignore it. Use the following three pokes.

POKE 58550,54
POKE 58551,15 - Paper/Ink
POKE 60641,3 - Border Colour

Thats all for this time, back soon...

2x2= TABLE FUN 5x3=
6x4= 4x9=
11x3= 2x5=
4x5= 12x12= 3x8= 6x2= 9x7=

By: Anne Wilson.

Educational software for young children is an area which is really neglected. A simple program can give hours of fun and test a childs abilities at the same time.

TABLE-FUN is just such a game, simple, easy to use but fun. I haven't wasted effort on flashy graphics or music that clutter up the screen and distract attention from the job in hand. If you want to add extras I think you will find the program easy to amend. Full instructions are contained in the program so I wont repeat them here. It can be used by children from quite a young age as all they have to do is recognise the correct answer and press a button

```
10 DIM s$(32)
20 DIM c(13)
30 LET f=0: LET v=0: LET sc=0: LET j=0: LET x=0: LET m=0: LET
e=0: GOTO 100
40 LET k=1: LET z=x: LET f=1: RETURN
50 LET sc=sc+1: PRINT AT 13,12; PAPER 1; INK 7;"CORRECT"
60 FOR i=1 TO 12: LET c(i)=0: NEXT i: RETURN
70 PRINT AT 11,0; PAPER 5;b;"X table";: PRINT TAB 22; PAPER 5;
"Speed ";s: FOR n=0 TO 7: PRINT AT 5,(4*n);n+1: NEXT n: OVER 1:
PRINT AT 5,0;"": OVER 0: RETURN
100 BORDER 5: PAPER 7: BRIGHT 1: INK 0: GOSUB 4000
110 PRINT AT 5,0;"Which table are you testing?": INPUT b
120 PRINT AT 5,0;s$;AT 5,0;"How fast? (1=slow, to 4=fast)": IN
PUT s$: IF CODE s$<49 OR CODE s$>52 THEN GOTO 120
125 LET s=VAL s$: IF s<0 OR s>4 THEN GOTO 120
130 LET f=0: LET u=0: LET k=0: LET z=100
140 PRINT AT 15,0;"Score ";sc
150 IF j=10 THEN GOTO 500
160 LET v=0: FOR i=1 TO 100: NEXT i: CLS
170 LET a$="": LET a=INT (RND*13): GOSUB 60: GOSUB 70
180 PRINT AT 2,0; PAPER 6;a;" X ";b;AT 15,0; PAPER 7;"Score ";s
c: PAUSE 120/s
190 FOR n=0 TO 28 STEP 4
200 LET i=INT (RND*13)+1: IF c(i)<>0 THEN GOTO 200
210 LET c(i)=1: LET x=(i-1)*b
220 IF x=a*b THEN GOSUB 40
230 PRINT AT 6,n; PAPER 5;x;: LET a$=a$+CHR$ x
240 PAUSE 50/s: NEXT n
250 LET j=j+k
260 FOR t=1 TO 300/s
270 IF INKEY$<>" " AND CODE INKEY$>47 AND CODE INKEY$<58 THEN LE
T v=VAL INKEY$: GOTO 300
280 IF INKEY$=" " THEN NEXT t
290 NEXT t
```

TASWORD +2

By: John Wase.

Taswords 3 and +2 are two of the programs currently giving DISCiPLE & PLUS D users the most trouble because of their non-standard shadow ROM calls. INDUG hopes to announce soon the launch of a full disc conversion of Tasword 128/+2, complete with OPENTYPE files, mailmerge etc. It is hoped that Tasword 3 will follow before the end of the year. But meanwhile Ian Burnage of Luton has pointed out that Tasword+2 uses many of the standard hook codes, and that it is therefore relatively easy to modify the standard cassette version to work on the DISCiPLE / PLUS D, though with a number of the more advanced features missing. In fact it needs a little mod to the basic and just four POKES to get a version which works.

The first versions of Tasword+2 showed up problems on many machines: a lot of keys, notably shifted seven and eight (the up & down cursor keys) refused to work for an hour or more after the +2 was switched on. Later versions of Tasword+2 contained a patch which solved the problem at the expense of features like keyboard buffering and print spooling. I assume this is the cassette you have and that you want to keep the dreaded patch. So just follow these instructions.

1. Load the tape. It loads a piece of Basic, then the main code, then throws up a screen. Break at this point.

2. Modify lines 10 and 80 as follows:

```
10 CLEAR VAL "25299": PRINT "" TAB VAL "7": "Loading Taswor
d": LOAD D1"tascode" CODE: CLS
80 CLS: LOAD D1"tasbas2"
```

And then SAVE D1"tasbas1" LINE 10

3. Execute the following POKES to the code:

```
POKE 51966,207: POKE 51967,49: POKE 36894,7: POKE 36897,17
```

4. Enter NEW as a direct command (your machine code is quite safe above the lowered RAMTOP) and MERGE the next piece of Basic from tape.

5. Modify line 10 so that it reads:

```
10 POKE @6,1: RANDOMIZE USR VAL"51966": STOP
```

And then SAVE D1"tasbas2" LINE 1

(Note the new address to run the code from).

```
300 IF v>8 THEN LET v=0
305 IF v=0 THEN GOTO 350
310 IF CODE a$(v)=a*b THEN GOSUB 50: GOTO 130
320 IF v<>0 THEN PRINT AT 13,12; INK 7; PAPER 2;"WRONG": IF f=1
THEN PRINT AT 13,10;"It was ";a*b: LET e=e+1: PAUSE 150: GOTO 1
30
330 PRINT AT 14,6; PAPER 7; INK 2;"It was not there!": LET e=e+
1: PAUSE 150: GOTO 130
350 IF z<>100 THEN PRINT AT 13,9; PAPER 6;"You missed ";z: PAUS
E 150: LET m=m+1 360 GOTO 130
500 FOR i=6 TO 21: PRINT AT 13,i;" ";AT 14,i;" ": NEXT i: FOR i
=9 TO 22: PRINT AT 11,i;"*";AT 16,i;"*": NEXT i
510 FOR i=12 TO 15: PRINT AT i,9;"*";AT i,22;"*": NEXT i
520 PRINT AT 13,15-(LEN n$/2);n$;"s";AT 14,11;"GAME OVER"
530 PRINT AT 18,10;"You missed ";m;".";AT 19,7;"You made ";e;"
mistake"+("s" AND e<>1);"."
540 COPY SCREEN$
550 PAUSE 0: RUN
600 IF CODE n$>96 AND CODE n$<123 THEN LET n$(1)=CHR$ (CODE n$(
1)-32)
610 FOR i=2 TO LEN n$-1: IF n$(i)<>" " THEN NEXT i
620 IF i<LEN n$ THEN PRINT ;: IF CODE n$(i+1)>96 THEN LET n$(i+
1)=CHR$ (CODE n$(i+1)-32): RETURN
630 RETURN
3999 STOP
4000 CLS : PRINT AT 1,11;"Table Fun": OVER 1: PRINT AT 1,11;"__
": OVER 0
4010 PRINT AT 4,0;" You will be shown a table sum.Then you will
see 8 possible answers. You choose the right one by enteri
ng 1, 2, 3, 4, 5, 67 or 8.";" HOLD DOWN the number until you s
ee it is understood."" If the right answer is not there ju
st wait for the new set."
4020 RANDOMIZE : INPUT "Please type your name and press
ENTER: ";n$: GOSUB 600
4030 CLS : RETURN
9900 SAVE d1"TABLE-FUN" LINE 10: VERIFY d1"TABLE-FUN"
```

SMALL ADS. SMALL ADS.

UTILITIES for sale:- Tasword 128 £8, Last Word £8,
Tasword 2 & Tasprint £3, Beta Basic 4 £8, Devpac 4 £8,
Spectre Mac/Mon £8, Picturesque Ed/Ass £4, Morse Reader £1
All originals with manuals. Ring Ted Oakley on 021-476-7974.

YOUR ADVERT

Buying, Selling, Pen Friends, Clubs, etc.

This space is reserved for you. Any PRIVATE advert, up to 30 words (subject to acceptance), will be printed FREE in the next available issue. Any software sold must be original copies, with full instructions and in working order. The publishers will not be held, in any way, responsible for adverts in this column.

6. RUN to initialise Tasword, change drive number to 1, do any other program customising you want, go into Basic from the main menu.

And then SAVE D1"tascode" CODE 25300,40236.

You can now load, save, merge and print text. You will have to go into Basic to catalogue your disc (though you can select disc 1 or 2 direct from the main menu), and you cannot print with datamerge or save Tasword from the menu option. The files produced are microdrive files and you will not transfer them successfully to another disc if you try to "MOVE" them.

Nevertheless you will have a working version with all the main functions in just 4 POKES, which should keep you going until the full disc version becomes available.

My thanks to Ian Burnage for working out the pokes.

* - * - * - *

BACK ISSUES

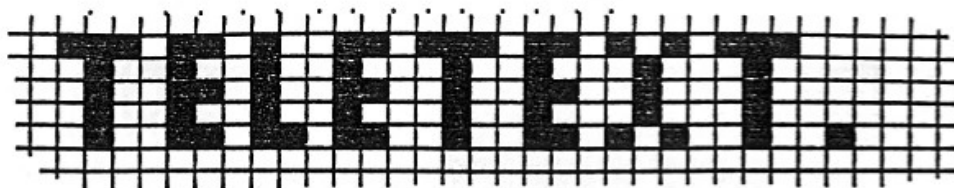
For members who have missed past issues of FORMAT (or perhaps worn theirs out through constant use) we run a back-issue service.

The cost is 65p per issue (85p overseas) which includes postage. Your copies will be sent out as soon as possible but, in order to keep printing costs down, it may take up to 28 days for us to dispatch. Make cheques, drawn on UK bank (or Euro-Cheques, cash, P.O.) payable to INDUG.

Available Issues

All issues are available starting with Issue #1 - August 1987.

Please WRITE YOUR ORDER ON A SEPARATE PIECE OF PAPER. DO NOT include correspondence with orders and remember your membership number or orders will be delayed.



By: P.Anderson.

Teletext is broadcast along with the television picture on all BBC and ITV channels. On BBC 1 and BBC 2 the system is called CEEFAX, ITV and Channal 4 call theirs ORACLE. But have you ever wondered how?

How can both a TV signal and the teletext pages be transmitted without interfering with each other? The secret lies in the FBI. No, not the American FBI its the TV FBI I'm talking about. To explain, a normal picture is made up of a series of horizontal lines, in Britain its 625 lines to be exact. But where is teletext? Well each frame of the picture is drawn by an electron beam. This starts drawing the picture at the top left hand corner. By the time it completes painting in one frame, the beam ends up pointing at the bottom right hand corner of the tube. Before starting on the next frame of the picture, it has to move itself from the bottom right to the top left. During this time, it must not draw anything, otherwise a diagonal stripe would appear across the screen. The process of moving the beam back to its starting position is called the Field Blanking Interval or FBI.

During this time, the action of the TV tube must be suspended, or blanked, even if the TV signal is still sending information. In fact, the time taken during the FBI is about equivalent to that taken to send up to 25 TV picture lines. As these lines are not used for the TV picture, they are the ideal place to store the teletext information. Some of these extra lines (about six on each channel at the moment) are used to carry the teletext signal which is transmitted in binary form.

As the lines were designed to carry a picture. it is a simple process to code a line to carry the binary data. If the line is divided into equal length sections, each unit can represent one bit. If the section is black, then it represents a zero, if its white it represents a one. In this way, a single television picture line is split up into 360 sections, which represent 45 eight bit bytes. Most TV sets take less than 25 lines for the FBI so you can actually see these teletext data lines at the top of the normal TV picture if you adjust the vertical height control to pull the top of the screen down.

How does a TV set cope with this? Well, a normal set (without teletext capability) simply treats them as it would any other picture line. Any lines received during the FBI are effectively ignored and any received between the end of the FBI and the top of the real picture are off the top of the screen so you dont see them. If on the other hand its a teletext set then circuitry inside the receiver detects and decodes these lines. This circuitry is set to scan all the 25 lines transmitted during the FBI. It will detect which lines carry a teletext signal by the

first three bytes on the row. If these decode to rubbish it will ignore the rest of the line. If, however, it detects a specific predefined pattern of white and black (representing the binary sequence 10101010 10101010 11100100) then it knows it has found a teletext signal and must decode the rest of the line. The reason for the alternating 1 and 0 pattern of the first two bytes is that these bytes are used by the decoder to 'sync' with the transmission. They allow it to determine the size of each section of the line representing a single binary digit. ? The purpose of the next byte is to tell the decoder where each byte starts. This byte is always the same number, and as it always appears immediately after the sync bytes, the sequence can be used to determine the start and end points of each byte.

All it now does is to break the rest of the line into chunks equivalent in length to a single bit. It then groups these in units of eight. This leaves 42 bytes on each TV line, which are used by the decoder to form the teletext display. The first two are used to identify the row as belonging to a particular page, and each of the remaining 40 bytes represents a character for display. There are just 40 bytes of data on each television line. This directly relates to the length of a single row of text on a teletext display (40 characters). As a teletext page requires 24 rows of text, this will take 24 television lines for its transmission. As only six TV lines can be used per transmitted picture frame, a single teletext page can be transmitted with every four frames. This may seem a long time until we realise that a complete TV picture frame is transmitted every 1/50th of a second, so a teletext magazine can be broadcast at the rate of 12.5 pages per second. It takes on average around 15-20 seconds to transmit every available page on a single channels magazine. The speed of transmission is nearly seven million bits per second (actually 6.9375 Mbits/s) but as we have seen, the system can only use six of the 625 television picture lines.

The BBC 2 Ceefax service has long had a Micro Magazine Area (Page 700) but this has been on the wain over the last six months. The News Page (701) once had 15-20 frames each week, now you're lucky to get 5 or 6. There also used to be a list of addresses for clubs and user groups (I got INDUG's address from it last year) but this has now disappeared. The Tele-Software pages, software stored for down-loading if you have the right interface, had always been BBC's related but now seems to be filling with IBM software. Last year Oracle, on Channel 4, was advertising a Teletext interface for the Spectrum, but I have never seen one (or even a review of one) so I don't know if they ever saw the light of day. I hope someone will soon come up with a cheap interface, with 780k per disc it would be possible to grab and store a vast amount of data from the transmitted pages. Just think, your own on-line Micronet...

With modern TV technology it will soon be possible to send 8 or even 12 lines per frame but what if we could forget about the TV picture and use all 625 lines for teletext. Before we had so much talk about 24 hour TV there was an idea to use the full screen, after normal hours that is, for bulk teletex transmission, but now? Well we will have to wait and see.

FORMAT SOFTWARE SERVICE

FORMATs Software Service provides DISCiPLE and PLUS D owners with a growing range of software specially designed for their systems. Software is supplied on tape for easy transfer to any format/size of disc.

Code	Title	Program Description
T001	LCOPY	Routines to replace the GDOS printer dump routines in your system file. Enables Epson compatible printers without Esc * to be used with both SCREEN\$ 1 & 2 and SNAPSHOT prints.
T002	MULTI-POKE	Program to display and edit 48K SNAPSHOT files. Full facilities for entering published POKES i.e. Infinite Lives etc. Makes use of printer if attached.
T003	G-HACKER	A graphic investigator. Look inside commercial programs, find the Sprites and pictures. Works in 48K mode but will handle many 128K programs.
T004	I.B.U.	The highly successful 'Incremental Backup Utility', written by Nev Young. As featured in Issues 5,6,7 & 8 of FORMAT.
T005	TAPE-SNAP	Transfer 48k Snapshots to tape with this easy to use program. Transferred Snaps will reload and run without the disc system present.
T006	ART STUDIO	Converts the OCP ART STUDIO to full disc operation, includes printer driver.
T007	PLUS D HACKER	For <u>PLUS D ONLY</u> . EXAMIN memory or program registers at any point using snapshot button, DISASSEMBLE, SEARCH graphics, enter POKES, return to running program + MANY MORE. 128k and printer fully supported. Includes Graphics Animator system extensions to basic. Why waste money on extra hardware, PLUS D HACKER does it all.

All tapes: £3.95 incl p&p or £4.95 overseas. Send order (on a separate piece of paper), clearly stating Tape Number; Title; Quantity required and your membership number. Payment in STERLING by Cheque (drawn on a UK bank), P.O.s, Euro Cheques or Cash accepted.

DO NOT ENCLOSE CORRESPONDENCE WITH ORDERS.